Asymmetrical Vertical Federated Learning

Yang Liu[1*] , **Xiong Zhang**[1] , **Libin Wang**[1]

[1]Tencent Cloud Product Department, Tencent, Shenzhen, China

{clarkieliu, farleyzhang,benlbwang}@tencent.com

Abstract

Federated learning is a distributed machine learning method that aims to preserve the privacy of sample features and labels. In a federated learning system, ID-based sample alignment approaches are usually applied with few efforts made on the protection of ID privacy. In real-life applications, however, the confidentiality of sample IDs, which are the strongest row identifiers, is also drawing much attention from many participants. To relax their privacy concerns about ID privacy, this paper formally proposes the notion of asymmetrical vertical federated learning and illustrates the way to protect sample IDs. The standard private set intersection protocol is adapted to achieve the asymmetrical ID alignment phase in an asymmetrical vertical federated learning system. Correspondingly, a Pohlig-Hellman realization of the adapted protocol is provided. This paper also presents a genuine with dummy approach to achieving asymmetrical federated model training. To illustrate its application, a federated logistic regression algorithm is provided as an example. Experiments are also made for validating the feasibility of this approach.

1 Introduction

Machine learning algorithms play an essential role in extracting patterns and useful knowledge from datasets. Traditional machine learning methods are usually developed from the centralized or parallel perspective, i.e., the overall datasets are stored in one, or a group of computing machines in hand, and the learning objective can be efficiently achieved by the machines with their direct access to the datasets. Although many of those algorithms are well-studied and have excellent performance over even petabyte-scale datasets, they are still confronted with serious troubles when implemented in the business or research fields with high concerns on data confidentiality. For example, every day portable and wearable smart devices collect billions of users' motion and body condition information, which contains numerous valuable patterns that benefit reproductions. However, it is impossible to

perform traditional machine learning algorithms for knowledge extraction, because the distributedly collected data are prohibited to upload to central servers subject to strict privacy laws and regulations, such as the GDPR. Similar examples also exist in a variety of areas, such as finance, health, education, etc.

Federated learning, first proposed in 2016 [McMahan *et al.*, 2016], is a privacy-preserving machine learning approach to analyzing the patterns of distributed sensitive datasets. The notion was later expanded to a clearer and more comprehensive framework with three components: horizontal federated learning in which participants share the same feature space, vertical federated learning in which participants share the same ID space, and federated transfer learning for disjoint data distributions. In this framework, every distributed participant shares encrypted messages that are computed based on their individual datasets and updates its local model according to the received messages, with or without the presence of a trusted third party. However, in federated learning, especially in the vertical scenario, it is common but unreasonable to omit the investigation of the sample ID's information leakage after leveraging them to align distributed datasets. In a standard vertical federated learning system, for example, samples held by different participants are aligned by executing some secure protocol and letting everyone know what the exact intersect set is. In practical applications, the participants are usually companies or institutions in competitive relations, and many clients of one company are potential advertising targets of another. On the one hand, thus, the participants would not allow the disclosure of their sample IDs. On the other hand, there are asymmetrical federations in real life, in which a subset of the participants are small companies with strong requirements of ID privacy protection, while the other are large companies who do not concern much ID privacy, because their guests are almost all citizens in the society. Such an unbalanced setup requires a vertical federated learning system to distinguish the "weak" and the "strong" sides of the federation and take into consideration their specific privacy protection demands.

In this paper, we specify the vertical federated learning into two classes: the symmetrical and the asymmetrical, to develop federated learning algorithms that preserve the privacy of sample IDs for the participants who indeed demand. The contributions of this paper are summarized as follows.

(i) We formally propose and comprehensively characterize

---

*Contact Author

the notion of asymmetrical vertical federated learning.

(ii) We incorporate the standard private set intersection protocol to achieve the asymmetrical ID alignment phase in an asymmetrical vertical federated learning system. In addition, we provide a Pohlig-Hellman realization of the adapted private set intersection protocol.

(iii) We present a genuine with dummy approach to achieving asymmetrical federated model training. To illustrate its application, we provide a federated logistic regression algorithm as an example. Experiments are also made for validating the feasibility of the approach.

The rest of the paper is organized as follows. In Section 2, we formulate the symmetrical and asymmetrical classification of vertical federated learning. We further present an asymmetrical private set intersection protocol for ID alignment and one of its realizations in Section 3. Section 4 provides the genuine with dummy approach to asymmetrical model training and its application to federated logistic regression, along with a few experimental validations in Section 5. Finally, several concluding remarks are given in Section 6.

2 Problem Definition

In this section, we revisit the notion of vertical federated learning and formally categorize it into two classes to study the privacy preservation of sample IDs.

2.1 Vertical Federated Learning

Let $\mathcal{D} = (\mathcal{I}, \mathcal{X}, \mathcal{Y})$ denote a complete dataset with $\mathcal{I}, \mathcal{X}, \mathcal{Y}$ representing the sample ID space, the feature space and the label space, respectively. It was defined in [Yang *et al.*, 2019a] that the vertical federated learning is conducted over two datasets $\mathcal{D}_1 = (\mathcal{I}_1, \mathcal{X}_1, \mathcal{Y}_1)$, $\mathcal{D}_2 = (\mathcal{I}_2, \mathcal{X}_2, \mathcal{Y}_2)$ satisfying

$$\mathcal{X}_1 \neq \mathcal{X}_2, \mathcal{Y}_1 \neq \mathcal{Y}_2, \mathcal{I}_1 = \mathcal{I}_2.$$

In real-world applications, however, it is nearly impossible to find two original datasets collected by distributed parties that share exactly the same sample ID space. Therefore, as a preparation phase for vertical federated learning, it is necessary to introduce appropriate ID-alignment protocols that assist each party with its secure identification of $\mathcal{I}_1 = \mathcal{I}_2$ and establishment of two datasets' row mapping. We define $\mathcal{D}_1^o = (\mathcal{I}_1^o, \mathcal{X}_1, \mathcal{Y}_1), \mathcal{D}_2^o = (\mathcal{I}_2^o, \mathcal{X}_2, \mathcal{Y}_2)$ as the pre-ID-alignment datasets that correspond to $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively. Clearly, $\mathcal{I}_1 \subset \mathcal{I}_1^o$ and $\mathcal{I}_2 \subset \mathcal{I}_2^o$. In addition, we write $\mathcal{I}^w$ as the whole ID space that contains all possible elements in $\mathcal{D}_1^o$ and $\mathcal{D}_2^o$. Then the vertical federated learning presented by [Yang *et al.*, 2019a] is depicted in Figure 1.

2.2 Private Set Intersection

Recall that ID-alignment protocols are essential in the preparation of vertical federal model training. To achieve ID alignment, Private Set Intersection (PSI) protocols, as one of the most well studied areas in secure multiparty computation, are usually implemented in a federated learning system. In standard PSI, each party $\mathcal{P}_i$ holds a set S_i, which involves $\mathcal{P}_i$'s confidential data. All parties would like to cooperatively find the intersection $S' = \bigcap_i S_i$, and in the meantime, each $\mathcal{P}_i$
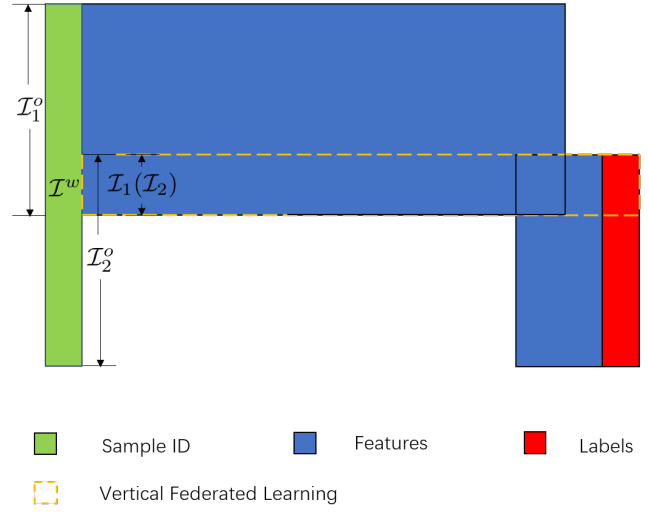

Figure 1: Symmetrical Vertical Federated Learning

keeps the elements in $S_i \setminus S'$ private. The realization of PSI protocols can be based on classical public-key cryptosystems [Freedman *et al.*, 2004; Liang and Chawathe, 2004], oblivious transfer [Pinkas *et al.*, 2014], garbled circuits [Huang *et al.*, 2012], etc.

2.3 Symmetrical and Asymmetrical

Let us call the parties $\mathcal{P}_1, \mathcal{P}_2$ who own $\mathcal{D}_1^o, \mathcal{D}_2^o$. As can be seen from Figure 1, $\mathcal{P}_1$ and $\mathcal{P}_2$ have equally powerful positions in the sense that the numbers of elements in their pre-ID-alignment sample ID spaces $\mathcal{I}_1^o, \mathcal{I}_2^o$ share the same order of magnitude and both sets play a major role in the whole ID space $\mathcal{I}^w$. In fact, even if $\mathcal{P}_1$ picks $e \in \mathcal{I}^w$ uniformly at random, there is $|\mathcal{I}_2^o| / |\mathcal{I}^w| \gg 0$ probability of $e \in \mathcal{I}_2^o$. In such "federation of the strong", each party would not gain much knowledge by executing PSI protocols to obtain the intersection $\mathcal{I}_1 = \mathcal{I}_2$, and thus it bothers neither party to let the other party know what IDs it possesses. Since finally both parties symmetrically obtain the intersection for model training, we term this scenario symmetrical vertical federated learning.

The opposite scenario is that $\mathcal{P}_2$, without loss of generality, has $|\mathcal{I}_2^o| / |\mathcal{I}^w| \approx 0$, while $\mathcal{P}_1$ still holds $|\mathcal{I}_1^o| / |\mathcal{I}^w| \gg 0$. It is common that the federated learning system to be established is for $\mathcal{P}_2$'s learning task, and thus the labels are provided by $\mathcal{P}_2$. This vertical data distribution is as shown in Figure 2. Clearly, there is a sufficiently small probability that an arbitrary ID $e \in \mathcal{I}^w$ one picks uniformly at random exactly belongs to $\mathcal{I}_2^o$. Therefore, we say $\mathcal{P}_2$ is at the weak side in the sense that each sample ID in $\mathcal{I}_2^o$ is regarded as sensitive information, whose privacy would be severely compromised through the revelation of $\mathcal{I}_1 = \mathcal{I}_2$ by executing standard PSI protocols. In such "federation of the weak and the strong", it is necessary to asymmetrically protect the ID privacy of the weak party in the ID alignment phase. This scenario is termed asymmetrical vertical federated learning.

It is reasonable not to analyze the "federation of the weak" scenario, because we may shrink the whole space $\mathcal{I}^w$ to $\mathcal{I}^w \setminus (\mathcal{I}_1^o \cup \mathcal{I}_2^o)$ so that this scenario can be reduced to the

"federation of the strong" one. Throughout this paper, we impose the following assumption.

Assumption 1 $\mathcal{I}^w = \mathcal{I}_1^o \cup \mathcal{I}_2^o$.

It is evident that if Assumption 1 holds, either $|\mathcal{I}_1^o| \geq \frac{1}{2}|\mathcal{I}^w|$ or $|\mathcal{I}_2^o| \geq \frac{1}{2}|\mathcal{I}^w|$. Now by representing a positive number n by $n = a \cdot 10^b$ with $a \in [10^{-\frac{1}{2}}, 10^{\frac{1}{2}})$ and naming b the order of magnitude of n, we provide the following precise definition for.

Definition 1 *Under Assumption 1, vertical federated machine learning can be classified into two categories based on the participating parties' pre-ID-alignment sample ID spaces.*

(i) It is called Symmetrically Vertical Federated Learning (SVFL) if

$$-\frac{1}{2} \leq \log_{10}\frac{|\mathcal{I}_1^o|}{|\mathcal{I}^w|}, \log_{10}\frac{|\mathcal{I}_1^o|}{|\mathcal{I}^w|} \leq 0.$$

(ii) It is called Asymmetrical Vertical Federated Learning (AVFL) if either one of the following two inequalities holds.

$$\log_{10}\frac{|\mathcal{I}_1^o|}{|\mathcal{I}^w|} < -\frac{1}{2}, \tag{1}$$

$$\log_{10}\frac{|\mathcal{I}_2^o|}{|\mathcal{I}^w|} < -\frac{1}{2}. \tag{2}$$

In AVFL, we further name $\mathcal{P}_1$ (or $\mathcal{P}_2$) the weak participant if (1) (or (2)) holds, and $\mathcal{P}_2$ (or $\mathcal{P}_1$) the strong participant.

Since Assumption 1 holds in Definition 1, there always exist a weak and a strong participant in AVFL. Note that we base the analysis above and the rest of the paper on the two-party federation case for the simplicity of demonstration. In fact, the analysis can be trivially extended to the multi-party case by asymmetrically protecting the sample ID privacy of the weak party.

3 Asymmetrical ID Alignment

ID alignment is typically the first stage of a vertical learning workflow. In this section, we adapt the standard PSI protocol to achieve asymmetrical ID alignment and provide a realization using a classical cryptosystem.

3.1 Asymmetrical PSI Protocols

Recall that in SVFL's ID alignment phase, the execution of standard PSI protocols not only lets each party gain knowledge about the samples they both hold, but also provides a tag that manages to efficiently link together distributed sample fragments and paves the way for the follow-up federated model training phase. To realize a proper ID-alignment phase in AVFL, one has to adapt standard PSI protocols so that they satisfy

(i) The exact intersection set $\mathcal{I}_1 = \mathcal{I}_2$ is kept private against the strong participant.

(ii) In the federated model training phase, the distributed samples with ID in $\mathcal{I}_1 = \mathcal{I}_2$ are still alignable.
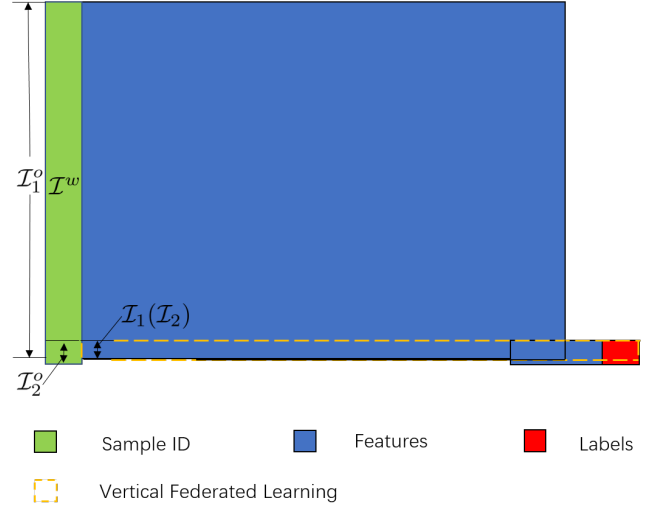

Figure 2: Asymmetrical Vertical Federated Learning

Let $\mathcal{I}_w^o, \mathcal{I}_s^o$ denote the pre-ID-alignment sample ID sets held by the weak and the strong participant, respectively. Alternatively, they are defined by

$$\{\mathcal{I}_w^o\} = \operatorname*{argmin}_{\mathcal{I}\in\{\mathcal{I}_1^o,\mathcal{I}_2^o\}} |\mathcal{I}|, \ \{\mathcal{I}_s^o\} = \operatorname*{argmax}_{\mathcal{I}\in\{\mathcal{I}_1^o,\mathcal{I}_2^o\}} |\mathcal{I}|.$$

A possible ID-alignment approach that meets requirements (i)-(ii) above is to use a variant of PSI protocols that we present below.

Definition 2 *Asymmetrical PSI (APSI) protocols, as a variant of PSI protocols, yield an obfuscated set $\mathcal{I}^{obf} \subset \mathcal{I}^w$ at each party satisfying*

$$\mathcal{I}_1 = \mathcal{I}_2 \subset \mathcal{I}^{obf} \subset \mathcal{I}_s^o.$$

In addition, only the weak participant further knows $\mathcal{I}_1 = \mathcal{I}_2$.

The output of such PSI protocols is illustrated in Figure 3.

3.2 A Pohlig-Hellman Realization

We now provide a realization of APSI protocols based on the Pohlig-Hellman encryption. Define $\mathbb{Z}_n = \{0, 1, \ldots, n-1\}$ and $\mathbb{Z}_n^* \subset \mathbb{Z}_n$ as the set of integers coprime to n. Let $G_n = (\mathbb{Z}_n^*, \cdot)$ denote a multiplicative group. Then the well-known Pohlig-Hellman encryption scheme is described by the following three components.

(i) (Key generation) Select a prime number p such that every plaintext is an element of G_p, where $p - 1$ has at least one large prime factors. For example, select $p = 2q + 1$ with q being also a prime number. Then select $a \in G_{p-1}$ and compute $a^{-1} \in G_{p-1}$. Finally, one reveals G_p as public knowledge and keeps (a, a^{-1}) as the key to this symmetrical cryptosystem.

(ii) (Encryption) Encrypt the plaintext $m \in G_p$ by

$$E_a(m) = m^a.$$

(iii) (Decryption) Decrypt the ciphertext $c \in G_p$ by
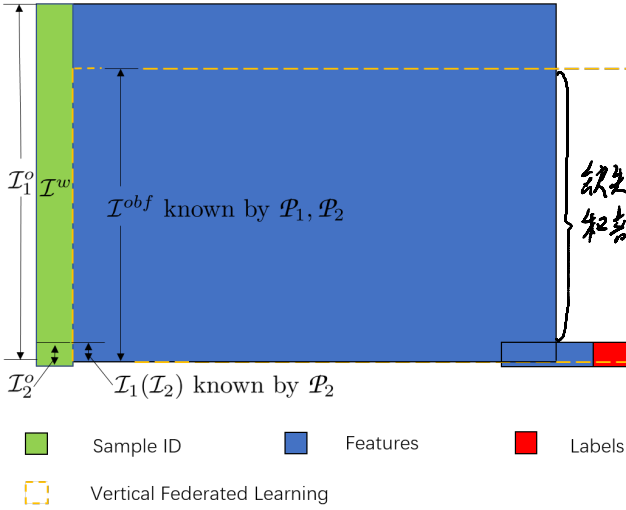
$$D_a(c) = c^{a^{-1}}.$$

Figure 3: Asymmetrical PSI Protocols for AVFL

It is straightforward to show the commutative property of the Pohlig-Hellman encryption by checking $E_a \circ E_b = E_b \circ E_a$ for any $a, b \in G_{p-1}$. Now we present an APSI protocol based on the Pohlig-Hellman encryption in Algorithm 1.

Security Analysis. In Algorithm 1, the exchanged information by step 6 is $U_1, U_2, U_2^\dagger$, whose security is evidently guaranteed by the Pohlig-Hellman encryption scheme. By step 8, the strong participant $\mathcal{P}_1$ receives the encrypted obfuscated set $U^{obf\dagger}$ but it cannot distinguish which elements belong to the encrypted true intersection set. In fact, $\mathcal{P}_1$ cannot even obtain any plaintext based on $U^{obf\dagger}$ because the elements in $U^{obf\dagger}$ are still encrypted with $\mathcal{P}_2$'s key b. By step 11, since the cooperatively decrypted message is only $\mathcal{I}^{obf}$, $\mathcal{P}_1$'s private information $\mathcal{I}_1^o \setminus \mathcal{I}^{obf}$ is protected against the weak participant $\mathcal{P}_2$. Clearly, in step 12, only $\mathcal{I}^{obf}$ is revealed to $\mathcal{P}_1$ with $\mathcal{I}_1^o \cap \mathcal{I}_2^o$ kept private by $\mathcal{P}_2$.

It can be noted that the security number λ is interpreted as a cardinality ratio $h(\lambda)$ of the obfuscated intersection set $\mathcal{I}^{obf}$ to the true intersection set $\mathcal{I}_1 = \mathcal{I}_2$. As can be directly computed, there is $1/h(\lambda)$ probability that a uniformly random element in $\mathcal{I}^{obf}$ that $\mathcal{P}_1$ picks belongs to $\mathcal{I}_1 = \mathcal{I}_2$. Clearly, $h(0) = 1$, in which case $\mathcal{I}^{obf} = \mathcal{I}_1 = \mathcal{I}_2$ and $\mathcal{P}_1, \mathcal{P}_2$ both obtain the true intersect set. In this special case, AVFL becomes SVFL. As λ goes up from zero to one, $1/h(\lambda)$ exponentially decreases, and thereby it becomes more difficult for $\mathcal{P}_1$ to potentially identify $\mathcal{I}_1 = \mathcal{I}_2$ from $\mathcal{I}^{obf}$. When λ reaches the maximum one, the obfuscated set $\mathcal{I}^{obf} = \mathcal{I}_1^o$, i.e., the whole ID space of $\mathcal{P}_1$ is used for obfuscating and $\mathcal{P}_1$ cannot gain any knowledge by executing Algorithm 1.

4 Asymmetrical Federated Model Training

In this section, we investigate the asymmetrical federated model training process and propose a novel and general approach to training a model in an asymmetrical fashion that is as good as in a symmetrical fashion. We also provide an application of this approach to an existing federated learning algorithm.

Algorithm 1 Pohlig-Hellman APSI Protocol

Input: The strong participant $\mathcal{P}_1$ holds $\mathcal{I}_1^o$. The weak participant $\mathcal{P}_2$ holds $\mathcal{I}_2^o$. Security number $\lambda \in [0, 1]$.
Output: $\mathcal{P}_1$ only obtains $\mathcal{I}^{obf}$ satisfying $\mathcal{I}_1^o \cap \mathcal{I}_2^o \subset \mathcal{I}^{obf}$ and

$$\frac{|\mathcal{I}^{obf}|}{|\mathcal{I}_1^o \cap \mathcal{I}_2^o|} = h(\lambda) \triangleq \left(\frac{|\mathcal{I}_1^o|}{|\mathcal{I}_1^o \cap \mathcal{I}_2^o|}\right)^\lambda.$$

$\mathcal{P}_2$ obtains both $\mathcal{I}_1^o \cap \mathcal{I}_2^o$ and $\mathcal{I}^{obf}$.

1: $\mathcal{P}_1$ and $\mathcal{P}_2$ negotiate a public group G_p for Pohlig-Hellman encryption.
2: $\mathcal{P}_1$ and $\mathcal{P}_2$ privately generates keys $(a, a^{-1}), (b, b^{-1}) \in G_{p-1} \times G_{p-1}$, respectively.
3: $\mathcal{P}_1$ and $\mathcal{P}_2$ computes $U_1 = \{E_a(x) : x \in \mathcal{I}_1^o\}$ and $U_2 = \{E_b(x) : x \in \mathcal{I}_2^o\}$, respectively.
4: $\mathcal{P}_1$ and $\mathcal{P}_2$ exchange U_1, U_2.
5: $\mathcal{P}_1$ and $\mathcal{P}_2$ computes $U_2^\dagger = \{E_a(x) : x \in U_2\}$ and $U_1^\dagger = \{E_b(x) : x \in U_1\}$, respectively.
6: $\mathcal{P}_1$ sends $U_2^\dagger$ to $\mathcal{P}_2$.
7: $\mathcal{P}_2$ randomly selects $U^{obf\dagger}$ satisfying $U_1^\dagger \cap U_2^\dagger \subset U^{obf\dagger} \subset U_1^\dagger$ and $|U^{obf}| = h(\lambda) |U_1^\dagger \cap U_2^\dagger|$.
8: $\mathcal{P}_2$ sends $U^{obf\dagger}$ to $\mathcal{P}_1$.
9: $\mathcal{P}_1$ computes $U^{obf} = \{D_a(x) : x \in U^{obf\dagger}\}$.
10: $\mathcal{P}_1$ sends U^{obf} to $\mathcal{P}_2$.
11: $\mathcal{P}_2$ computes $\mathcal{I}^{obf} = \{D_b(x) : x \in U^{obf}\}$ and $\mathcal{I}_1^o \cap \mathcal{I}_2^o = \mathcal{I}_2^o \cap \mathcal{I}^{obf}$.
12: $\mathcal{P}_2$ shares $\mathcal{I}^{obf}$ to $\mathcal{P}_1$.

4.1 Genuine with Dummy Approach

Using APSI protocols given in Definition 2, the weak participant obtains the true intersection set $\mathcal{I}_1 = \mathcal{I}_2$, which is a subset of the obfuscated intersection set $\mathcal{I}^{obf}$ that the strong participant knows. As shown in Figure 3, the vertical federated learning domain now contains margins, i.e., all labels and a few features are missing for the samples with their ID in $\mathcal{I}^{obf} \setminus \mathcal{I}_1$. Indeed, federated transfer learning [Yang *et al.*, 2019a] can be used to fill in the margins by the feature-representation-transfer approach [Pan and Yang, 2009]. Nevertheless, in many practical applications, such as financial risk management, it is normal to train machine learning model based on the small but exactly original data in $\mathcal{I}_1 = \mathcal{I}_2$, in order to avoid misjudgment of dishonest conduct. In these areas, the learned features and labels for $\mathcal{I}_{obf} \setminus \mathcal{I}_1$, even if not "negatively transferred", may lead to undesirably strict or loose risk control strategies. Therefore, it is necessary to design asymmetrical model training schemes that take the distributed output of APSI protocols and yield the same or almost the same result as the SVFL. Based on the standard vertical model training in [Yang *et al.*, 2019a], we now present a Genuine with Dummy (GWD) approach to achieving asymmetrical model training as follows. Note that a trusted third party $\mathcal{P}_3$ is introduced as a secure coordinator.

(i) $\mathcal{P}_3$ generates a public-key cryptosystem, and sends the public key to $\mathcal{P}_1, \mathcal{P}_2$.

(ii) $\mathcal{P}_1, \mathcal{P}_2$ exchange intermediate variables to cooperatively

compute gradient and loss. The weak participants, say $\mathcal{P}_2$, normally execute computation rule for the samples in $\mathcal{I}_1$, i.e., the genuine, but set the variables that correspond to the samples in $\mathcal{I}^{obf} \setminus \mathcal{I}_1$, i.e., the dummy, to specific mathematical identities so that their existence will not affect the relevant computed result. The identities can be, for example, zero in addition, one in multiplication or $f(x) = x$ in function composition.

(iii) $\mathcal{P}_1, \mathcal{P}_2$ turn to $\mathcal{P}_3$ for gradient and loss decryption service, and update their local models.

As we can see, the central idea of the GWD approach provided above is to let the weak participant execute the normal protocol for the genuine samples, while mathematically mute the dummy samples that it actually does not hold before sending them to the strong participant. To keep the strong participant unaware of the existence of the dummy samples, a potential method is to implement semantically secure encryption scheme, such as Paillier cryptosystem [Paillier, 1999], to disable the participant from efficiently distinguishing the identities out of a group of normal variables. In addition, it is clear that the asymmetrical model training achieved with the GWD approach would exhibit exactly the same performance as the standard (or symmetrical) model training because the introduced identities strictly guarantee invariant intermediate results at every step.

An illustration of GWD architecture is provided in Figure 4, which depicts the federated execution process of a general subroutine. The weak participant would like to send encrypted messages v_1, v_2 to the strong participant and then expect a response of the execution result $subroutine(v_1, v_2)$. However, the direct transmission would give away the genuine IDs. Instead, the weak participant sends v_1, v_2 along with the mathematical identities e_1, e_2 corresponding to the dummy samples, and finally receives $subroutine(v_1, v_2, e_1, e_2)$, which is identical to what it expects. Besides, the strong participant performs the computation $subroutine(\cdot)$, while remaining oblivious to the genuine IDs.

4.2 Asymmetrical Vertical Logistic Regression

We now take the coordinator-free federated logistic regression training presented in [Yang *et al.*, 2019b] as an example and adapt it to an asymmetrical training protocol using the GWD approach. Let $\langle M \rangle$ denote the ciphertext of a plain message M. Then the adapted Asymmetrical Vertical Logistic Regression (AVLR) training protocol is presented in Algorithm 2.

In Algorithm 2, the central subroutine is to let the weak participant $\mathcal{P}_2$ share the encrypted scalars $\langle \phi_{ik} \rangle$, based on which the strong participant $\mathcal{P}_1$ evaluates its local gradient $\langle \nabla^s \mathcal{L}(\mathbf{w}_k) \rangle$. To implement the GWD approach, $\mathcal{P}_2$ computes ϕ_{ik} for $e_i \in \mathcal{I}_1^o \cap \mathcal{I}_2^o$ based on the true labels, but set $\phi_{ik} = 0$ for all $e_i \in \mathcal{I}^{obf} \setminus (\mathcal{I}_1^o \cap \mathcal{I}_2^o)$, which is the addition identity. Due to the fact that $crypto$ preserves semantic security, $\mathcal{P}_2$ can neither identify $\langle 0 \rangle$ from the received set of ϕ_{ik}s, nor distinguish which samples are the dummies. Since $\mathcal{P}_1$ performs additions to compute $\nabla^s \mathcal{L}(\mathbf{w}_k)$, the presence of the dummy samples' ϕ_{ik} would have no effects on the result.

Algorithm 2 Asymmetrical Vertical Logistic Regression Model Training

---

Input: The strong participant $\mathcal{P}_1$ holds the sample set $(e_i, \mathbf{x}_i^s) \in \mathcal{I}^{obf} \times \mathbb{R}^{m^s}$. The weak participant $\mathcal{P}_2$ holds the sample set $(e_i, \mathbf{x}_i^w, y_i^w) \in \mathcal{I}_1^o \cap \mathcal{I}_2^o \times \mathbb{R}^{m^w} \times \{0, 1\}$. Learning rate $\eta > 0$.

Output: $\mathcal{P}_1, \mathcal{P}_2$ learn the weights $\mathbf{w}_*^s \in \mathbb{R}^{m^s}, \mathbf{w}_*^w \in \mathbb{R}^{m^w}$, respectively, such that the joint weight $\mathbf{w}_* = (\mathbf{w}_*^{s\top}, \mathbf{w}_*^{w\top})^\top$ is the global optimum of the model.

1: $\mathcal{P}_2$ instantiates a semantically secure additively homomorphic cryptosystem $crypto$.

2: $\mathcal{P}_1$ and $\mathcal{P}_2$ initialize their model estimates $\mathbf{w}_0^s \in \mathbb{R}^{m^s}$ and $\mathbf{w}_0^w \in \mathbb{R}^{m^w}$, respectively.

3: **while** $k = 0, 1, 2, \ldots$ **do**

4:    $\mathcal{P}_1$ sends $\mathrm{L}_k^{obf} = \{(e_i, \mathbf{w}_k^{s\top}\mathbf{x}_i^s) : e_i \in \mathcal{I}^{obf}\}$ to $\mathcal{P}_2$.

5:    Based on L_k^{obf}, $\mathcal{P}_2$ computes $l_{ik} = \mathbf{w}_k^{s\top}\mathbf{x}_i^s + \mathbf{w}_k^{w\top}\mathbf{x}_i^w$ for all $e_i \in \mathcal{I}_1^o \cap \mathcal{I}_2^o$, and further

$$\phi_{ik} = \begin{cases} y_i - \left(1 + \exp(-l_{ik})\right)^{-1} & \text{if } e_i \in \mathcal{I}_1^o \cap \mathcal{I}_2^o; \\ 0 & \text{if } e_i \in \mathcal{I}^{obf} \setminus (\mathcal{I}_1^o \cap \mathcal{I}_2^o). \end{cases}$$

6:    $\mathcal{P}_2$ uses $crypto$ to encrypt ϕ_{ik} for all $i \in \mathcal{I}^{obf}$ and sends $\mathrm{F}_k^{obf} = \{(e_i, \langle \phi_{ik} \rangle) : e_i \in \mathcal{I}^{obf}\}$ to $\mathcal{P}_1$.

7:    $\mathcal{P}_2$ computes the global likelihood and its local gradient in plaintext space

$$\mathcal{L}(\mathbf{w}_k) = \frac{1}{|\mathcal{I}_1^o \cap \mathcal{I}_2^o|} \sum_{e_i \in \mathcal{I}_1^o \cap \mathcal{I}_2^o} y_i l_{ik}$$
$$- \log(1 + \exp(l_{ik})),$$
$$\nabla^w \mathcal{L}(\mathbf{w}_k) = \frac{1}{|\mathcal{I}_1^o \cap \mathcal{I}_2^o|} \sum_{e_i \in \mathcal{I}_1^o \cap \mathcal{I}_2^o} \phi_{ik} \mathbf{x}_i^w.$$

8:    Based on F_k^{obf}, $\mathcal{P}_1$ computes its local gradient in cipherspace

$$\langle |\mathcal{I}_1^o \cap \mathcal{I}_2^o| \cdot \nabla^s \mathcal{L}(\mathbf{w}_k) \rangle = \sum_{e_i \in \mathcal{I}^{obf}} \langle \phi_{ik} \rangle \mathbf{x}_i^s,$$

9:    $\mathcal{P}_1$ further masks its local gradient by $G^s = \mathbf{r}_k \odot \langle |\mathcal{I}_1^o \cap \mathcal{I}_2^o| \cdot \nabla^s \mathcal{L}(\mathbf{w}_k) \rangle$, where $\mathbf{r} \in \mathbb{R}^{m^s}$ is randomly chosen and $\odot$ denotes the Hadamard product. $\mathcal{P}_1$ further sends G^s to $\mathcal{P}_2$ for decryption.

10:   $\mathcal{P}_2$ uses $crypto$ to decrypt G^s and divides the result by $|\mathcal{I}_1^o \cap \mathcal{I}_2^o|$. Finally, $\mathcal{P}_2$ manages to send back $\mathbf{r}_k \odot \nabla^s \mathcal{L}(\mathbf{w}_k)$ to $\mathcal{P}_1$.

11:   $\mathcal{P}_1$ uses the Hadamard division to recover $\nabla^s \mathcal{L}(\mathbf{w}_k)$.

12:   $\mathcal{P}_1$ and $\mathcal{P}_2$ update their local models by

$$\mathbf{w}_{k+1}^s = \mathbf{w}_k^s + \eta \nabla^s \mathcal{L}(\mathbf{w}_k),$$
$$\mathbf{w}_{k+1}^w = \mathbf{w}_k^w + \eta \nabla^w \mathcal{L}(\mathbf{w}_k),$$

    respectively.

13: **end while**

	genuine_id_1	v_1
	genuine_id_2	v_2
	dummy_id_1	e_1
	dummy_id_2	e_2

v_i : intermediate variable
e_i : identity

$subroutine(v_1, v_2, e_1, e_2)$
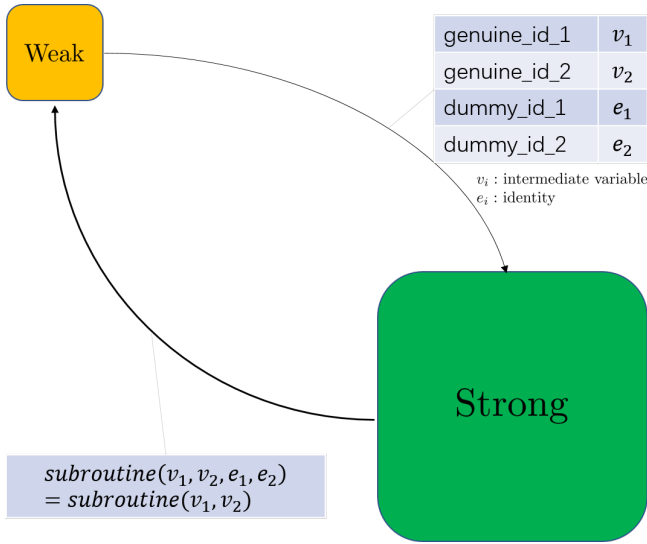$= subroutine(v_1, v_2)$

Figure 4: Architecture of the GWD approach

In this way, the dataset, over which the federation of $\mathcal{P}_1$ and $\mathcal{P}_2$ train their model, is inherently the sample set with IDs in $\mathcal{I}_1^o \cap \mathcal{I}_2^o$. This guarantees that the asymmetrical vertical training yields the same result as the standard (symmetrical) vertical training does.

5 Experiments

This section provides a few experiments that validate the feasibility of the APSI + AVLR protocol and compare it with the existing standard (symmetrical) protocol.

5.1 Settings

We implement our APSI and AVLR algorithms in the federated learning framework FATE[1]. The performance of the APSI + AVLR protocol is demonstrated over the dataset MNIST[2], which has 60000 samples and 784 features. To adapt the dataset to our distributed setup, we manually allocate an ID to each sample, split and assign the partitions to the weak participant $\mathcal{P}_2$ and the strong participant $\mathcal{P}_1$ as in Table 1.

	392 Features	392 Features
10000 Samples	$\mathcal{P}_2$	$\mathcal{P}_1$
50000 Samples	(Abandoned)	

Table 1: MNIST Dataset Partitioning

As indicated by Table 1, the underlying federated training would be performed over the 10000 samples that both $\mathcal{P}_1$ and $\mathcal{P}_2$ hold. Therefore, the training performance is expected not to be as good as other algorithms that take the whole dataset as input. However, it is fairly reasonable to utilize the distributed setup in Table 1, since the experiments are conducted for the purpose of validating the feasibility of the

---

[1]https://github.com/FederatedAI/FATE
[2]http://yann.lecun.com/exdb/mnist/

APSI + AVLR protocol and comparing it with the standard (symmetrical) version.

For the computing hardware, we use two individual machines to serve as the participants and either of them has 4 CPU cores and 16GB RAM. These machines are both located in the same region of Tencent Cloud[3].

5.2 Numerical Results

In the experiments of the APSI + AVLR protocol, we adopt the fixed learning rate $\eta = 0.15$ but various security numbers $\lambda = 0, 0.25, 0.5, 0.75, 1$, and let the training process execute for 150 iterations. It is worth mentioning the standard (symmetrical) model training corresponds the $\lambda = 0$ case in our experiments. We plot the trajectories of training loss and AUC in Figure 5 and Figure 6, respectively. In Figure 5, the trajectories are almost the same for different λs, especially for the $\lambda = 0$ and the $\lambda \neq 0$ case, which also holds true for the AUC trend in Figure 6. The observation of these figures validates that the APSI + AVLR protocol has as good performance as its symmetrical version due to the introduction of mathematical identities.
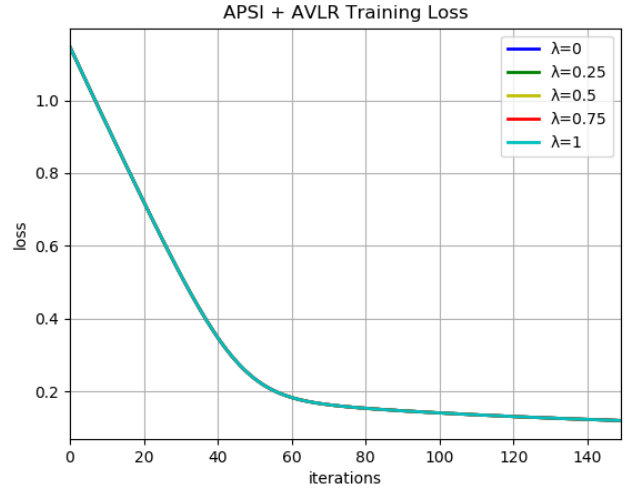

Figure 5: Training Loss Trajectories of the APSI + AVLR Protocol

6 Conclusions

In this paper, we studied the privacy preservation of sample IDs in vertical federated learning. To meet the privacy protection demands of different participants, we first proposed the notion of asymmetrical vertical federated learning. We then adapted the standard private set intersection protocol to achieve the asymmetrical ID alignment phase in an asymmetrical vertical federated learning system. Correspondingly, a Pohlig-Hellman realization of the adapted protocol was provided. To achieve asymmetrical federated model training, we also presented a genuine with dummy approach. We illustrated its application by providing a federated logistic regres-
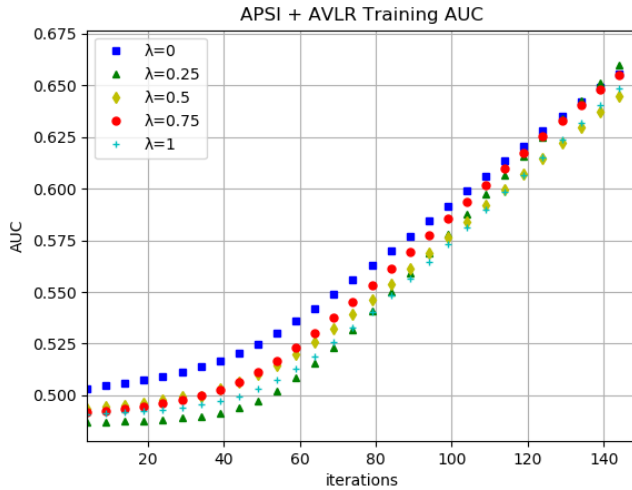
---

[3]https://cloud.tencent.com/

Figure 6: Training AUC Trajectories of the APSI + AVLR Protocol

sion as an example. Experiments were also made for validating the feasibility of this approach.

References

[Freedman *et al.*, 2004] Michael J Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 1–19. Springer, 2004.

[Huang *et al.*, 2012] Yan Huang, David Evans, and Jonathan Katz. Private set intersection: Are garbled circuits better than custom protocols? In *NDSS*, 2012.

[Liang and Chawathe, 2004] Gang Liang and Sudarshan S Chawathe. Privacy-preserving inter-database operations. In *International Conference on Intelligence and Security Informatics*, pages 66–82. Springer, 2004.

[McMahan *et al.*, 2016] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629*, 2016.

[Paillier, 1999] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 223–238. Springer, 1999.

[Pan and Yang, 2009] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2009.

[Pinkas *et al.*, 2014] Benny Pinkas, Thomas Schneider, and Michael Zohner. Faster private set intersection based on {OT} extension. In *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pages 797–812, 2014.

[Yang *et al.*, 2019a] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[Yang *et al.*, 2019b] Shengwen Yang, Bing Ren, Xuhui Zhou, and Liping Liu. Parallel distributed logistic regression for vertical federated learning without third-party coordinator. *arXiv preprint arXiv:1911.09824*, 2019.